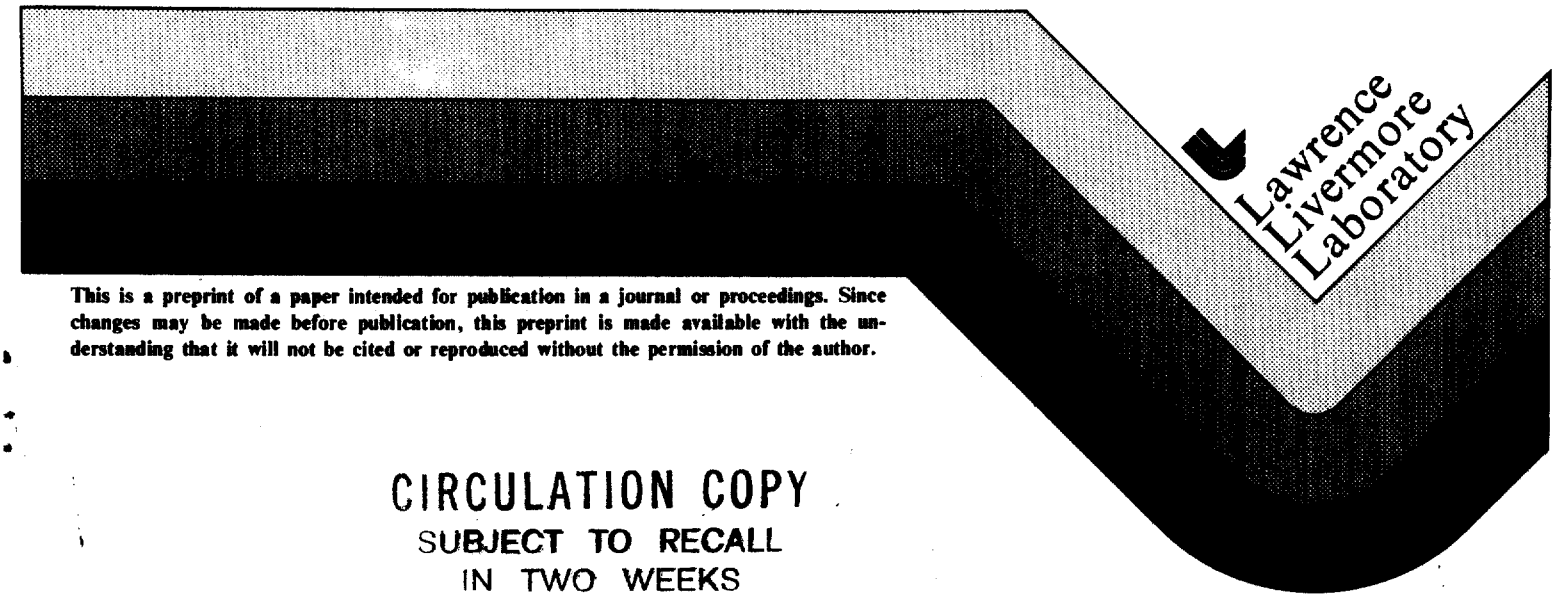
UCRL-84569
PREPRINT

USER SERVICES IN A NETWORK ENVIRONMENT

Jay M. O'Dell
Arthur B. Scott

This Paper was Prepared for Submittal to
Eighth Annual User Services Conference
November 16 - 19, 1980
Morgontown, West Virginia

June, 1980



USER SERVICES IN A NETWORK ENVIRONMENT *

Jay O'Dell and Arthur Scott
National Magnetic Fusion Energy
Computer Center
Lawrence Livermore National Laboratory
P.O. Box 5509
Livermore, California 94550
Telephone (415) 422-1545

Abstract

User services in a network environment are quite different from those in environments with users in close proximity. Communication with remote users with advanced degrees requires different attitudes, knowledge, and techniques. The problems and solutions of education, documentation, software tools, and helping users debug programs over long distances will be presented.

INTRODUCTION

A computer center whose users are spread throughout the United States has some unique problems compared to computer centers whose users are physically nearby. Problems relating to the education of users, the dissemination of documentation, user program preparation, and program debugging become unique in our situation because they must be dealt with at long distance. It is not the intention of this paper to prescribe overall solutions, but to present some of the methods used to deal with the problems at this computer center.

THE NATIONAL MAGNETIC FUSION ENERGY COMPUTER CENTER

The National Magnetic Fusion Energy Computer Center (NMFECC) is located at the Lawrence Livermore National Laboratory in Livermore, California (approximately 40 miles east of San Francisco). The U.S. Department of Energy (DOE) created the computer center to provide large scale computing capability for its contractors working on the magnetic confinement of plasmas to produce fusion energy. The computer user community is distributed over the continental United States.

HARDWARE TOOLS

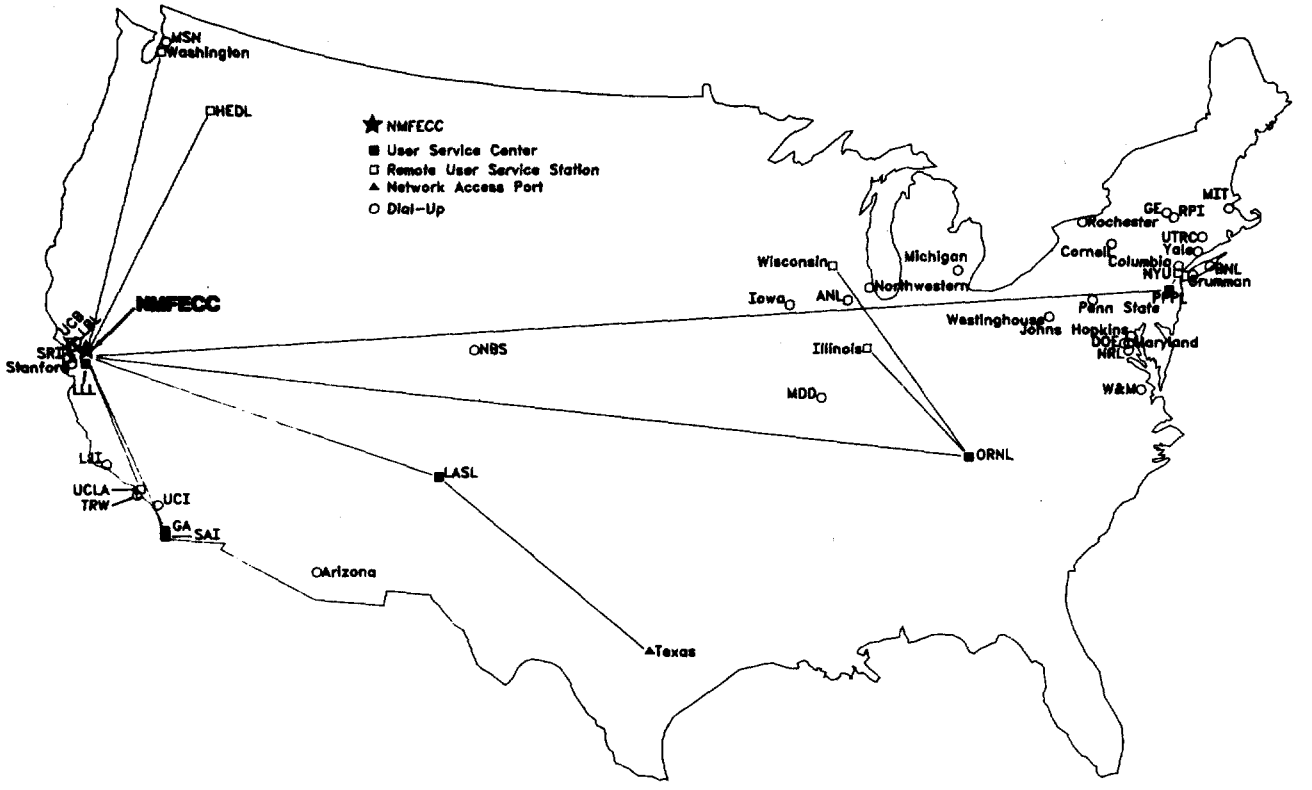
The NMFECC has two "super" computers, a CDC 7600 and a Cray-1. There are about 1300 users in the NMFECC user community. They have access through network connections and through an extensive dial-up capability located at the central site. The NMFECC has several different types of network connections. Six midi-computers (DEC PDP-10's) at different geographical locations are connected to the computer center through fifty-kilobit lines. An additional "in-house" PDP-10 is used by the computer center to test and develop software for the network. In addition to accessing the large computers the PDP-10's are used for smaller scale problems not requiring the large computers. Also on the network are seven Remote User Service Stations (RUSS's) and one Network Access Port (NAP). A RUSS consists of a printer with both alphanumeric and graphics capability. It also serves as a concentrator for up to sixteen user terminals. The NAP is connected to a VAX11-780 which is used in a manner similar to the PDP-10's. The NMFECC also supports connections to two other networks, ARPANET and TYMNET. Except for periods of maintenance and system development, the computer center operates twenty-four hours per day, every day of the year. The center distributes user output (film, microfiche and paper) throughout the country by mail.

PURPOSE

Since the mandate of the National Magnetic Fusion Energy Computer Center is to provide a large scale computer facility for those people doing research in magnetic fusion energy for DOE, the scope of the network is geographically vast, and it is always in a state of continued growth. For example, the center has users distributed across four time zones, located in Washington, Wisconsin, Massachusetts, New York, New Jersey, Tennessee, Washington, D.C., New Mexico, Arizona, and California to name a few states. Users come from private industry, universities and national laboratories. Typically, the user is a physicist, engineer, technician, or computer scientist. These computer

*Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore Laboratory under contract number W-7405-ENG-48.

NATIONAL MFE NETWORK



scientists help with the local computer development and are not employees of the Magnetic Fusion Energy Computer Center at Livermore. The education, background and responsibilities of users vary greatly.

The primary purpose of a centralized computing facility is to provide the highest quality in large scale computing in the most economical way.[1] The centralization of hardware also facilitates the sharing of software between users who are working on similar projects. In order to make this resource easily usable, the NMFECC maintains its own timesharing operating systems on its large computers. Interactive access (as opposed to batch access) is the key to providing service to users. Hard or softcopy software documentation is accessible via an online document program. Microfiche is also available. Besides the normal methods of communication (telephone, U.S. Mail), two-way communication between the center and its users is provided through online MAIL and NEWS programs. The center also publishes a monthly newsletter. The documentation access program is an interactive program that prompts the user for which documents to display at the terminal, or to print off-line. The "view" mode allows the user to view portions of a document by keyword at his(her) terminal. (This feature comes in handy when users do not have a hardcopy document at their side. e.g., when they are working at home.) In "print" mode users can dispose documents to a network printer, an off-line high speed printer, or an off-line microfiche recorder. The latter two forms are then mailed to the user.

CONSULTING TOOLS

The document program can teach the novice user how to use itself via three levels of instructional prompts: terse, more verbose, and most verbose. The user can select the different levels at will, or the document program will change levels after the user has made a few mistakes. This feature allows the "greenest, fresh off the street" user to obtain documentation.

Given the above distribution design, we have taken some pains in regards to the contents of documents. A novice user is usually overwhelmed by the sheer volume of material that is available on any computer system. We have created a number of tutorial documents to get people started. In our tutorials we describe in detail the events that need

to take place when a novice user is first getting started. (We also try to carry this tutorial idea over to documents that are not intended for the novice user). If a picture is worth a thousand words, one example is more valuable. Indeed, when you do not give examples you are committing the crime of bewilderment.[2]

The advantages of our documentation system are threefold. First, the documents are stored in machine readable form so they can be updated easily, second, users have instant access to all the current documents, and third, the printed documents can be processed and charged to the user like any other computer output. The obvious advantage of having a program to print hardcopy documents is the lack of human intervention. Trying to get started at any new installation can be particularly frustrating. Usually, the user has a source program, along with input data. The user must then order or borrow manuals in order to learn how to input the source deck and data cards, compile the program, execute the program and output the results. With the ease of obtaining documentation via the DOCUMENT program, getting started is less of a chore.

Work in magnetic fusion is expanding with the consequence that the NMFECC is continually acquiring more users. To save new users from unwittingly rewriting existing programs, we have developed a system whereby users can submit abstracts of programs they are willing to share. Other users may then interrogate the abstracts interactively to find what applications programs are already available.

A MAIL program exists to send messages from one user to another. The message can be formatted by a text editor or with the MAIL program and then passed through the MAIL program to the intended user. After the MAIL message is sent, the receiving user is notified that MAIL is waiting to be read. MAIL is an ideal vehicle for sending information to users who are unavailable by other communication methods.

In the same way that MAIL is user oriented, the NEWS program is site or division oriented. NEWS can be used to send messages to the entire user community or to a particular site. For instance, a user may want to tell everyone about a bug discovered in a utility program; therefore, he sends a global message. But a local message about program usage could be limited to the user's site or division. NEWS is monitored each working day by the computer center to determine what problems users are having. NEWS is also used by the center to announce machine schedules and to explain unscheduled downtimes.

We publish a monthly newsletter in which we describe the work being done by the staff. We also use the newsletter to propose plans and policies and solicit users' responses. The "operation's statistics" and schedules are also published in the newsletter.

Another useful system utility is GIVE. With this program, one user can give copies of files to another user. The GIVE program is mentioned here because it is often necessary for a consultant to examine a user's program before the consultant can help the user. Users can often be helped much more efficiently if they can give a copy of a file to the consultant so he(she) can examine it while talking to the user on the telephone. Also, the consultant can obtain the file, work out a solution, and return the corrected file with appropriate comments and messages to the user. The GIVE program transfers files immediately. There is no waiting for a tape to be read and written or cards to be punched.

Another way that we help new users to get started is by holding periodic classes here at the center. We try to break the classes up into two halves: one half for the beginning user, and the other half for the more experienced user. These classes generally last one week. The NMFECC consulting staff has also visited some of the larger sites to give seminars and classes.

Scheduled classes are important to the users, because it gives them a chance to apply new ideas to their production programs. Since the programs that users have written give reasonable answers, they are unwilling to change them, even if they are inefficient or exhibit poor programming practices. Classes give users time to experiment and apply new ideas to their everyday tasks. It is therefore beneficial to the longtime user to attend classes and to become acquainted with more sophisticated aspects of the operating system, libraries, or utilities which will aid him in his daily work.

SOFTWARE TOOLS

Of particular interest is the operating system used by the NMFECC. In the early 1960's the Lawrence Livermore National Laboratory decided to write the first timesharing operating system for a large mainframe. (This system has been bootstrapped to different large computers.) This operating system is used by two national laboratories, Lawrence Livermore National Laboratory

and Los Alamos Scientific National Laboratory. The operating system is called the Livermore Time Sharing System (LTSS). The system runs on the CDC 6600, CDC 6400, and the CDC 7600. Most recently, the NMFECC has converted LTSS to run on a Cray-1 computer where it has been christened the Cray Time Sharing System (CTSS). The system is interactive. Thus, users can dial up the NMFECC and log into any of our computers, (PDP-10, CDC 7600 or Cray-1). It follows that all users must become acquainted with LTSS before they can do any meaningful computations.

The NMFECC has many general and special purpose subroutine libraries available. The libraries are linked to the user's program at load time and perform the usual tasks such as linking with I/O devices, and communicating with the operating system. In addition, the libraries have other uses such as generating graphics and performing string manipulations. We mention the library structure because the human engineering of the libraries and the library documentation is invaluable in making advanced capabilities accessible to the average user.[1]

The NMFECC has other software tools such as compilers, loaders, preprocessors, text editors, etc. Most of the software tools are written with a minimal "HELP" package to remind the interactive user of the possible commands.[3] The NMFECC programmers try to keep their utility routines as "human-engineered" as possible. Lately, the center has been trying to standardize the command interface to all of its utility routines. For example, we are moving toward a uniform method for specifying input files, output files, output media, etc. Another example is a uniform filename subset selection syntax so that users can implicitly select sets of files for processing.[4] Each utility produces desired results when the correct input stream is typed, and the program also prints meaningful error messages when the program encounters an incorrect parameter. Emphasis is placed on messages being terse and meaningful. Program names are purposely descriptive to trigger an association to a desired result.

Possibly the most important tools available to the NMFECC consultant are the dynamic debugging tools. The debuggers can be used statically to do a postmortem examination of dead programs by displaying the contents of variables and registers, or by retracing the subroutine linkage upward from the point of error. The debuggers can also be used dynamically to "breakpoint" ones way through a program in order to look for early symptoms of catastrophic failure. Sometimes a catastrophic failure will destroy the evidence so that interactive debugging is the only way to find the bug. In the "bad old days", everything was done by batch processing, and the available debugging tools were the compiler listing, load map, and octal dump. The advent of the dynamic debuggers is a tremendous improvement.

THE USER SERVICES GROUP

Now that we have described the tools we have at our disposal, and the geographically dispersed user community, let us consider the user services group at the NMFECC. The group consists of ten people, six of whom are general consultants. A portion of the group is assigned to answer the telephones on a rotating basis. Usually, two people are assigned to the telephone each day. There are two answering devices to receive telephone messages when the consultants are not at their desks. The answering devices are necessary because the users on the east coast are at work three hours before the consultants in California arrive at the computer center. With the recording devices it is not necessary for the computer center to be manned with consultants before a question can be asked.

The consulting staff can usually answer the users' questions immediately. However, in particularly difficult cases we may need more time in which to solve the problem, or to confer with a person more experienced in the problem area. That person will then return the users call with the solution.

As stated earlier, the operating systems that are at Livermore are timesharing systems that were developed earlier by in-house programmers over many years. The systems work very well, although we admit, like other operating systems, they still harbor some as yet undetected anomalies. It is not our intent to talk about the merits or shortcomings of any particular system, but to present a picture of the resources the NMFECC's users have at their disposal. We believe we provide resources and tools that are more usable than would generally be available under the batch operating systems typically furnished by vendors of large computers. The consequence is that a new user's previous experience with the vendor's system is not transferable. We feel however that our documentation system (which itself takes advantage of the interactive operating system) does put the new user directly in touch with the needed information.

We naturally encourage users to learn about our system and its programming tools through our

documentation system. Once they have done so, their need for personal consultation is reduced to questions about why their programs do not work. In particular we often get questions about why a program has failed when only a "trivial" change has been made. At this point users simply want to know what is wrong. A good consultant will not debug the user's program, but will teach the user how to diagnose the problem and how to use the debugging tools. The consultant can try to deduce from the symptoms what must have changed so that the user is gently led into confessing the nontriviality of his changes. If the user denies any guilt, the consultant then advises the user on how to use one of the debuggers.

PROGRAM PROBLEM AREAS

Assuming that we have stable hardware and software, a user's problems tend to fall into one of the following categories.

1) Incorrect language usage.

2) Incorrect library subroutine usage (bad argument list)

3) Incorrect system utility usage.

4) Floating point error (or a calculation that goes to zero)

5) Overstore, jump out of bounds, or illegal instruction., e.g.

```
      DIMENSION A(100)
      DO 10 I=1,1000
   10 A(I) = 3.14159
```

Incorrect language use is not always detectable by the compiler since the use may be syntactically correct. For example

DO 10 I=1 10 obviously compiles to DO10I=110

Missing type-declarations or inconsistent common block usage can also cause trouble.[5]

Incorrect library usage can sometimes be detected by the library routine, but if it is not, the user can check the actual argument values with a debugger by breakpointing at the entry to the library routine.

Incorrect system utility usage leads to an unexpected result and usually a cry of outrage by the user. In this case, adequate communication can save the day.

Floating-point exceptions are detected by the hardware. A debugger can be used to retrace subroutine calls and display actual argument or variable values.

Overstores, jumps out of bounds, and illegal instructions are the most difficult problems to solve. It is usually necessary to breakpoint one's way through a program and continually look for evidence that something is going wrong.

Given all of these software tools and the excellent quality of the staff, we are still confronted with the problem of helping the greatest number of people and staying sane. Our philosophy is to provide both the novice and experienced users with all the tools they need to do their own work. The user should be able to create, debug, and use his own programs with a minimum of help from the consultants. Sometimes the situation arises where the system software has a bug rather than the user's program. In this case, the consultant will have to make a judgment on whether to look for a system error or advise the user to look for a user error. If the user has reached the end of his mental and emotional rope, then the consultant should look at the problem to get the user back on the right track.

SUMMARY

As for consultant's sanity, we mentioned that we rotate the consulting duty, and each consultant has programming assignments to keep his own skills up to date and to keep his attitude fresh. Good communication depends on a good frame of mind. If a person becomes jaded, his responses become automatic and probably very shallow. The user needs to know that someone at the computer center cares about his problems and that someone is reducing the difficulties that he has in getting his program going. The attitudes and solutions mentioned here help the user services function of the NMFECC, but the most important feature here is an intelligent, well trained consulting staff that knows not only the technical aspect of their work but is also skilled in interpersonal communication. The user services consultant must always remember that a user is using the computer as a tool to complete a specific task. The user has results that he needs from the computer; he does not worry about the problems of the computer scientist. The user operates under different circumstances than the computer scientist. The user has his deadlines and pressures and does not want to be concerned with the fallibility of software or hardware.

1. Martin D. Robbins, William S. Dorn, and John E. Skelton, "Who Runs The Computer", Westview Press, Inc., Boulder Co., 1975.

2. James Martin and Adrian R. D. Norman, "The Computerized Society", Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1970.

3. B. Kernighan and P. Plaugher, "Software Tools", Addison-Wesley Publishing Company, Menlo Park, Ca., 1976.

4. Herbert A. Simon and Laurent Silklossy, "Representation and Meaning", Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1972.

5. Glenford J. Meyers, "Software Reliability", John Wiley and Sons, New York, 1976.